|  |  |
| --- | --- |
| Ad-Hoc Digital Forensics | Pre-1984 |
| FBI Computer Analysis and Response Team | 1984 |
| IRS Seized Computer Evidence Recovery Specialist | 1985 |
| International Association of Computer Investigative Specialist | 1990 |
| 1st FBI Conference on Computer Evidence | 1993 |
| International Organization on Computer Evidence | 1995 |
| Defense Computer Forensic Laboratory | 1998 |
| FTK created | 1999 |
| 1st Digital Forensic Research Conference | 2001 |
| 1st Forensic Education Program Accreditation Commission | 2003 |
| Sleuth Kit & DoJ Electronic Crimes Reference Guide | 2008 |

Law enforcement DF – Prosecution | Military DF – Continuity of Ops | Business DF – Availability of Service DF Process: 1. Identify 2. Preserve 3. Collect 4. Examine 5. Analyze 6. Present Covert Channel: modulation of data in a manner that allows data to be indirectly transmitted and prevents detection and analysis File slack: remaining sectors to the end of cluster sector: 512 bytes RAM slack: slack between the end of the logical file and the sector

Order of Volatility: 1. CPU Registers & Cache 2. RAM Contents 3. Network Connections 4. Running Processes 5. Hard Drive File System 6. Removable Backup media

7 Layers of Abstraction: 1. Disk 2. Partition 3. File System 4. Data Unit 5. Meta data 6. File Name Admissible Evidence: conform to certain legal rules before it can be put before a court Authentic Evidence: positively tie evidentiary material to the incident

Complete Evidence: tell the whole story and not just a perspective Reliable Evidence: Collected and subsequently handled that casts doubt about its authenticity and veracity Believable Evidence: readily believable and understandable by the court NIST Categories: 0. Exercise – Testing 1. Unauthorized Access 2. DoS 3. Malicious Code 4. Improper Usage 5. Scan – Attempted Access 6. Investigation DoD Categories: 0. Exercise – Testing 1. Root Level Access 2. User Level Access 3. Unsuccessful Access Attempt 4. DoS 5. Improper Usage – Misconfiguration 6. Scan 7. Malicious Code 8. Investigation 9. Explained Anomaly Incident Response Policy: 1. Incident Identification 2. Incident Investigation 3. Incident Damage Repair 4. Documenting Organizational Response 5. Updating Incident Response Procedure Chain of Custody Steps: 1. Document all collected items 2. Identify collecting agent 3. Segregate items in containers 4. Calculate hashes of each item 5. Securely transport evidence 6. Conduct proper hand-off of evidence 7. Secure items when stored Sandboxing in regards to virtualization: separation of guest resources from external sources VM Clone: replica of VM for backup or operational purposes VM provision: allocation of a host resource for a guest Snapshot of VM: image capture of VM Logical Analysis of file system should include definitions for: disk, partition, file system HDD is made up of: heads, platters, tracks, and sectors HDD file deletions only unlink reference to data Data location on disk is derived utilizing Logical Block Addresses Given LBA: LBA \* 512 bytes/sector % 1073741824 bytes/gig /dev/ - device file repository dmesg: kernel level driver messages lsblk: list block devices blockdev: block device system calls lsscsi: list SCSI disk attributes lsusb: list usb buses df: list disk space hdparm : display and set drive parameters Max size of of FAT12 partition – 16 MiB

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| IntelMQ | Collecting and processing security feeds (such as log files) using a message queuing protocol | Python | CentOS, Debian, OpenSUSE, Ubuntu | Reduce complexity of sys admin  reduce complexity of writing new bots  provide easy way to create your block lists | Reliability, data harmonization, events can be lost in the process |
| LaikaBOSS | Object scanning system used for malware analysis and intrusion detection | Python | Scalable, Flexible, Verbose/transparent | Limited file type by default, hard to extract files | Limited file types by default hard to extract files  Produces unnecessary metadata |
| PhotoRec | Recovers files, ignores file system | C | Windows, Linux | Very quick in recovering files, ignores the file system so it will work if the system is damaged | Recovered files do not have the same file name  No directory structure |
| PowerForensics | All-inclusive framework for hard drive forensics analysis | Python | Windows |  |  |
| GRR | Incident response framework focused on remote live forensics | Python | Windows, Linux | Scalable, Remote, analytical Tool | Scalability comes with limitations, no stream captures, analytics must be done separate from state capture, can be used for malicious intent |
| Linux Expl0rer | Easy to use US for command line tools Remotely connect to linux endpoints for observations | Python | Linux | Test different YAY rules all in one place, automatically search public hashes on Virus total | More commands, easier to for API keys, easier setup for nginx |
| OSQuery | Allow user to ask questions about your linux for windows system | C++ Python Ruby Powershell Objective C++ | Linux windows MacOS RPM Debian | Flexible and powerful tool, gives valuable information about the system | Not very user-friendly, lack of UI, Documentation hard to navigate |
| Guymager | Forensic imager | C++ Typescript | Linux | Very quick, Multi-threaded | Cant id hidden areas, not the fastest, only works for linux |
| Bulk Extractor | Stream based forensic tools, scans media from start to end | C++ C GNU Flex | Linux MacOS Windows | Provides a simple data report  Produces a precise and uncomplicated histograms that conveniatelly displays the features of the data | Susceptible to crashes due to errors in C buffer processing |
| FLOSS | String Obfuscator | Python | Widows OSX Linux | Automation, Insight into functionality | Uses a brute force approach, may return false positives |
| Swap Digger | Automates swap extraction and search | BASH | Linux | Great forensic tool for potentially accessing critical user information | Only works on swap partition and files, may return false positives |
| InVtero.Net | Tool designed to assist in analyzing memory | C# Iron Python | Windows | Simplifies the method to isolate and extract process memory |  |
| KeeFarce | Extracts passwords from KeyPass | C/++/# |  |  |  |